

## PRAKTIKUM IZ PROGRAMIRANJA 1

### - domaći zadatak broj 4 -

Sastaviti program na programskom jeziku Pascal, koji radi sa tekst datotekama po sledećem redosledu izvršavanja:

- a) poziva potprogram koji učitava imena datoteka sa kojima će raditi,
- b) poziva odgovarajući potprogram koji izvrši traženu obradu nad datotekama,
- c) ponavlja prethodne korake sve dok korisnik za ime datoteke ne zada prazan string.

Zavisno od rednog broja problema, sastaviti program, koji obavlja sledeću obradu:

0. čita ulaznu datoteku, koja predstavlja promet na kreditnim karticama, gde je svaki red po formatu **brojKartice ime prezime promena** (broj kartice – string od tačno 7 znakova, ime i prezime – stringovi do 30 znakova, promena – realan broj) i koja može u susednim redovima imati podatke o istoj osobi (isti su ime i prezime); program u izlaznu datoteku prepisuje ime i prezime osobe, sve njene brojeve kartica i ukupnu promenu po formatu **ime, prezime, brojKartice1, ..., brojKarticeN, ukupnaPromena** (za svaku osobu u zasebnom redu); prva četiri znaka svakog broja kartice u izlaznoj datoteci zameniti sa \*;
1. čita ulaznu datoteku, koja predstavlja red letenja, gde je svaki red po formatu **idLeta polazište destinacija hh:mm** (identifikacija leta – ceo broj, polazište i destinacija – stringovi do 25 znakova, dok je poslednji podatak o vremenu trajanja leta u satima i minutima) i koja može u susednim redovima imati podatke o povezanim letovima (isti su destinacija prvog i polazište drugog); program u izlaznu datoteku prepisuje podatke o letovima koji mogu da se povežu po formatu **polazište, destinacija, ukupnoTrajanje, međutačka1 - ... - međutačkaN** (za svaki let u zasebnom redu);
2. čita ulaznu datoteku sa podacima o pristupima internet servisu, gde je svaki red po formatu **internetServis ipAdresa pocetak kraj** (internet servis – string do 20 znakova, ip adresa korisnika – string do 16 znakova, pocetak i kraj – vremena u formatu hh:mm:ss) i koja može u susednim redovima imati podatke o istom servisu i korisniku (isti su internet servis i ip adresa korisnika); program treba da napravi izlaznu datoteku čiji redovi sadrže informacije o ukupnoj dužini pristupa korisnika internet servisu po formatu **ipAdresa internetServis vreme** (za svakog korisnika u zasebnom redu);
3. čita ulaznu datoteku, koja sadrži podatke o turneji mužičkog sastava, gde je svaki red po formatu **država grad datum** (država i grad – stringovi do 30 znakova, datum je u obliku **gggg-mm-dd**); podaci su uređeni rastuće po datumu; sastav se ne vraća u već posećenu državu; program treba da napravi izlaznu datoteku po formatu **država, brojGradova, datumPočetka, datumKraja** (drugi podatak je broj gradova koji će biti posećen u datoj državi, datum početka i datum kraja su prvi i poslednji datum nastupa u datoj državi po formatu **dd.mm.gg**); u izlaznoj datoteci treba da se nalaze samo države u kojima će sastav nastupati u više od 2 grada, svaka država u zasebnom redu;
4. čita ulaznu datoteku, koja sadrži podatke o karting trkama, gde je svaki red po formatu **idKorisnika ime prezime mm:ss:ms brojKrug** (identifikacija korisnika – ceo broj; ime, prezime – string do 30 znakova; mm:ss:ms – vreme prolaza jednog kruga u minutima, sekundama i milisekundama, redni broj kruga – ceo broj) i koja može u susednim redovima imati podatke o istoj osobi (isti su identifikacija korisnika, ime i prezime); program u izlaznu datoteku štampa ime i prezime osobe, njena tri najbolja vremena prolaza i redne brojeve krugova u kojima su postignuta ta vremena **ime, prezime, mm:ss:ms1, ..., mm:ss:ms3, broj1, ..., broj3** (za svaku osobu u zasebnom redu); u slučaju da je osoba odvozila manje od tri kruga za vremena prolaza i redne brojeve krugova koji ne postoje upisati **xx:xx:xx** i **xx**.

Program treba da čita ulaznu datoteku uz vođenje računa o tipu podataka koji se čita. Za sve podatke koje ima smisla grupisati, osmisliti sopstveni tip zapisa i koristiti promenljive tog tipa, tamo gde je to moguće. Podatke iz datoteka nije dozvoljeno smeštati u bilo kakvu zbirku podataka u memoriji (na primer, niz). Smatrati da ulazna datoteka ima 0 ili više redova. Radi lakšeg čitanja, smatrati da svi stringovi sadrže tačno jednu reč. Ako nešto u postavci zadatka nije dovoljno precizno definisano ili ako su neki od zahteva međusobno suprotstavljeni, usvojiti razumnu pretpostavku i rešiti zadatak korišćenjem te pretpostavke. Osmisliti i napisati na papiru nekoliko primera ulaznih podataka sa kojima će se program testirati, kao i očekivani izlaz za odabrane primere. Test primere otkucati u bilo kom tekst editoru i priložiti zajedno sa tekstom programa. Potprogrami ne smeju pristupati promenljivama glavnog programa direktno, već samo putem svojih argumenata.

**Važno:** Po uspešnom rešavanju osnovnog zadatka, napraviti modifikaciju programa tako da program obrađuje samo one linije iz ulaznih datoteka čiji je redni broj u skupu vrednosti iz opsega [1,255], koje korisnik unosi sa tastature.

Napomene:

1. Odbrana četvrtog domaćeg zadatka je u utorak, 20.12. i sredi, 21.12.2011. godine. Tačni termini za sve studente su objavljeni u okviru sistema WebLab.
2. Formula za redni broj problema i koji treba rešavati je sledeća (R – redni broj indeksa, G – poslednje dve cifre godine upisa):  
$$i = (R + G) \bmod 5$$
3. Kao rešenje domaćeg zadatka potrebno je na odbrani pokazati sledeće datoteke:
  - `dz4.PAS`, koja sadrži izvorni tekst programa na programskom jeziku Pascal;
  - `dz4_mod.PAS`, koji sadrži modifikovani program koji radi sa skupovima;
  - `dz4_u1.txt`, ..., `dz4_uN.txt`, koje sadrže nekoliko odabranih test primera kojima će program biti testiran.

13.12.2011. godine

*sa predmeta*